

複数決定木の動的順序付けに基づく発話の概念推定

谷 垣 宏 一[†] 政 瀧 浩 和^{†,*} 匂 坂 芳 典[†]

音声翻訳における中間言語等で表現される発話内容の推定を目指し、相互参照機能を持つ決定木群による理解方式を提案する。発話内容(概念)は、その種類と可能な組合せを規定された複数の意味的な項の集合として表される。本方式では、発話内容を推定するために、各項の表出を推定する複数の決定木と、それらの動的な順序付けアルゴリズムを用いる。それぞれの決定木は、他の木への参照機能により他項の表出を考慮して推定を行う。このため、規定された組合せ制約のもとで最適な項の集合を推定することが可能となっている。英語対話コーパスを用いた実験により、各項を独立に推定する手法と比較して11%の誤り改善率が得られた。

An Algorithm for Concept Identification Using Decision Trees

KOICHI TANIGAKI,[†] HIROKAZU MASATAKI^{†,*}
and YOSHINORI SAGISAKA[†]

This paper proposes an efficient decision tree algorithm to identify the semantic content in an utterance, represented in a formal language such as interlingua. Semantic content in utterance (concept) is represented as a certain combination of terms predefined. Possible combinations of those terms are also predefined. The proposed method employs a set of decision trees, each of which infer the existence of a certain term in the utterance, while examining the existence of other terms with its referential links on the nodes. Moreover, an algorithm is proposed to drive those trees under a dynamic ordering scheme. With this algorithm we obtained the optimal combination of terms for given utterance, for certain predefined combinations. Experimental results showed 11% reduction in error rate, compared to the traditional method which employs mutually independent decision trees.

1. はじめに

自然な発話音声は音響的にも言語的にも自由度が高いため、その認識はきわめて難しい問題である。一方、音声認識を適用する音声対話を考えると、ディクテーションのような一部の応用を除き、一般に話者が伝達を意図する「文意」の存在を考えられることが多い。この場合、音声認識による単語列への復号化は、文意理解のための下位タスクと考えられ、システム全体は、文意理解の観点から最適化が図られて然るべきである。このように、音声認識・言語理解の過程を、音声から文意への識別問題と見なすこともできる。こうした視点に立つと、表層から文意への識別知識として抽出される単語と文意の間の結束性を、認識時の言語制約として用いることで、合目的な意味からの認識制約として利用が期待できる。本稿では、音声認識への意味

的拘束として用いるための文意理解方式を提案する。

従来、言語理解処理は、人手で文法を構築し統語・意味解析を行うことで実現されてきた。しかし、文法の構築には、タスク・文法記述両面からのエキスパート知識が必要であること、人手で文法を構築することによるコスト、タスク間ポータビリティがない、といった問題から、近年、確率統計モデルによる研究がさかんに行われている。

特に自然で自発的な発話では、倒置、省略、言い直しなど、文法から逸脱した表現が現れる。こうした自然発話特有の言語現象に対し、先験的知識から被覆率の面で最適な文法を構築することは難しい。また、文法の過生成は認識制約としての利用に問題を残す。そこで我々は、模擬会話により収集した自然発話データベースに文意を表す意味タグを付与したコーパスを作成し、実際の自然発話に近いデータで自動学習した統計理解モデルを構築する。これにより、自然な会話体表現に対する頑健な理解方式の実現と、認識制約として有効な知識の抽出が可能と考えられる。

本稿では、文意として中間言語の構成要素である発

[†] ATR 音声翻訳通信研究所

ATR Telecommunications Research Institute

* 現在、住友金属工業

Presently with Sumitomo Metals

話内容（概念）を対象とし、これを統計的手法に基づいて推定する方式を提案する。なお、本稿では単語列を対象とした、高精度な言語理解方式について検討を行うものとし、音声への適用については今後の課題とする。今後、たとえば、音声認識仮説群に対し文意尤度を評価することで再順序付けを行う方法や、あるいは、文意を対話状況管理に利用することで次発話認識用の言語モデル選択を行う方法^{15),16)}等により、音声理解率の向上を図ることが可能と考えられる。

以下、2章では我々が扱う中間言語について説明する。3章では、組合せ制約に起因する概念推定上の問題を明らかにする。4章では、この問題を解決する決定木アルゴリズムを提案する。5章では、提案アルゴリズムの評価実験について述べる。6章で、本稿のまとめと今後の予定について述べる。

2. 中間言語

文意の表現形式として、音声翻訳国際コンソーシアム CSTAR (Consortium for Speech Translation Advanced Research)⁴⁾が定める中間言語を用いる。本章では、この中間言語について説明する。

2.1 Dialogue Act

この中間言語で表現された文意を“Dialogue Act”（以下、DA）と呼ぶ。

DA は、タスクに依存して定義される。現在の対象タスクは、旅行会話のサブドメインであり、これには、ホテル案内・予約、観光ツアーの間合せ、交通手段の間合せ、等が含まれている。DA は、これらサブドメインにおいて、タスク達成に直接関係する情報の伝達を目的に設計されており、DA 定義にない話題等は扱わない（データベース中では、こうした発話には“no-tag”が付与されており、5章の評価実験では除外している）。

また、DA は、その意味表現単位として“Semantic Dialogue Unit”（以下、SDU）と呼ぶ、おおむね単文相当の単位を仮定している。したがって、DA は必ずしも元の発話と1対1に対応しない。長い発話は複数のSDUに分割され、それぞれのSDUからDAへの変換が行われる。ただし、本稿では発話からSDUへの分割は扱わない。以下では、事前にSDUに分割された部分発話の入力を仮定する。特に断らない限り、こうした部分発話を含めて、すべて発話と呼ぶことにする。

DA は多言語翻訳のための枠組みである。DA から目的言語への変換には、テンプレートによる文生成といった簡易な手法が想定されている。

表1 DAによる文意表現例
Table 1 Example of DA representation.

発話	<i>the week of twelfth we have both singles and doubles available</i>
Speech act	give-information
Concept	availability+room
Argument	room-type=(single & double), time=(week, md12)
発話	<i>and I already have a room reserved there from the eighth through the sixteenth</i>
Speech act	give-information
Concept	reservation+features+room
Argument	time=(start-time=md8, end-time=md16)
発話	<i>so I wonder if I can change my reservation to that from the queen to the double</i>
Speech act	request-action
Concept	change+reservation+features+room
Argument	change-from=(bed-type=queen), change-to=(bed-type=double)
発話	<i>when do you plan on arriving in pittsburgh</i>
Speech act	request-information
Concept	temporal+arrival
Argument	location=pittsburgh, time=question
発話	<i>and the carnegie science center is showing a great movie called stormchasers on their omnimax screen</i>
Speech act	give-information
Concept	features+event
Argument	location=carnegie.science.center, event=stormchasers, event-type=(movie, omnimax)

2.2 DAの構成要素

表1にDAによる文意の表現例を示す。イタリック体の部分が実際の発話である。DAでは文意を、Speech act, Concept, Argumentの3つの観点から表現する。

Speech actは、話者の意図情報であり、発話(SDU)が媒介する命題内容に対する話者の態度を表す。表1の1番目の例では、Speech actとして“give-information”が与えられている。現在、Speech actとしては41種類が定義されている。

Conceptは、発話(SDU)の中心となる概念を、複数の概念要素conceptの組合せで表現する。以下本稿では、個々の概念要素のことを「concept」と呼び、一方、conceptの組合せにより表現される発話の概念のことを「組合せconcept」または「Concept」と呼ぶこととし、注意深く区別して用いる。たとえば、表1の1番目の例において、組合せconceptは“availability+room”であり、これを構成するconceptは“availability”と“room”である。現在、conceptは85種類が定義されている。このうち、5章の実験データに出現する43種類のリストを表2に示す。

Argumentは、ホテルの名称や宿泊日数、地名と

いった, Speech act や Concept では表現されない詳細な情報を, argument-name への argument-value の割当て, および, それらの関係子による結合で表現する. 表 1 の 1 番目の例において, “room-type” と “time” が argument-name であり, “single”, “double”, “week”, “md12” が argument-value である. argument-name は, 現在 81 種類が定義されている.

2.3 組合せ制約

前節では, DA が, Speech act, Concept, Argument の 3 つレベルから表現されることを述べた.

ただし, これらの要素の任意な組合せが許されていないわけではない. Speech act と Concept, あるいは, concept 間の可能な組合せは定義として与えられる. この組合せ定義は, 表 3 にその一部を示すように, 階層的なリストとして与えられる. 左カラムは Speech act であり, これに組み合わせることが可能な Concept を右カラムに示している. concept 間の可能な組み合わせは, この表の右カラムに存在するものに限定される (また, ここでは示さないが, 表 3 の組合せに

対応して使用可能な argument-name の規定が存在し, さらに, 各 argument-name に対しても割当て可能な argument-value の規定が別に存在する).

この組合せ制約は, 主に文生成に用いるテンプレート構築の便宜を考慮して規定されたものである. 組合せの定義からは, タスク上無意味な組合せが排除されている. たとえば, “availability” は「(ある対象の) 利用可能性」を意味する concept である. この “availability” は, “room” や “transportation” といった concept と結合して, “availability + room” (「部屋の利用可能性」) や “availability + transportation” (「交通機関の利用可能性」) といった組合せ concept を作る事が許されるが (表 3 1~3 行目), 一方で, 「団体 (客)」 “party” との組合せ “availability + party” は許されていない. また, 文生成上, 有意な差を生じない組合せはあらかじめ 1 つにマージすることで, 組合せ数の縮退が図られている. たとえば, “reservation + features + room” (「ある特徴を有する部屋の予約」) という組合せ concept が定義されている一方で, “reservation + room” (「部屋の予約」) は定義リストに存在しない. これは, 後者の文テンプレートを前者と別に用意する必要がないためである.

現在, Concept (組合せ concept) の数は 22,344 種類が定義されている. また, Speech act と Concept の組合せでは, 132,264 種類が定義されている.

3. 概念推定モデル作成上の課題

以下, 本稿では, Concept (組合せ concept) のみを推定対象とする. これを Speech act との組合せとして推定することも可能ではあるが, Speech act は文脈に強く依存するため, 発話単位での最適化を図る手法に対しては正確な評価の妨げとなる. したがって, 本稿では扱わず, 対話状況利用を含めた今後の課題と

表 2 concept の一覧
Table 2 “concept” list.

arrival	availability	cancellation
change	checkin	checkout
confirmation	currency	departure
destination	event	exchange
expiration-date	fax-number	features
flight	guide	help
help-again	hotel	location
minimum-stay	name	numeral
party	payment	person
preference	price	price-difference
price-fluctuation	reservation	room
send	sight	spelling
stay	telephone-number	temporal
tour	transportation	trip
unavailability		

表 3 組合せ定義の一部
Table 3 Part of combination definitions.

Speech act	Concept		
give-information	availability		
	availability	+	room
	availability	+	transportation
	features		
	features	+	room
	features	+	transportation
	features	+	party
...			
request-action	reservation		
	reservation	+	features + room
	reservation	+	features + transportation
...			

する。また、Argument はおおむね Table lookup と local parse で対応可能と考えられるため、ここでは考慮しない。

Concept 推定モデルの作成においては、以下に述べる点を考慮することが必要である。2.3 節で述べたように、concept は組合せで 1 つの有意なまとまりをなすように設計されており、その可能な組合せは定義セットとしてアприオリに与えられる。発話に対応する組合せ concept は、定義セットに許された組合せの中で最適なものが与えられる。これは、単語と concept の間の、多対多のマッピングであり、モデリングにおいて concept 間の非依存性を仮定することは適切ではないと考えられる (3.3 節で具体例とともに詳述する)。

3.1 従来の統計的理解モデル

従来から、統計的アプローチによる言語理解の研究が行われている。

Kuhn ら²⁾は、Semantic Classification Trees (SCTs) と呼ぶ、一種の決定木を複数用いることで、発話から SQL への変換を行っている。しかし、SCT をデータベース項目および検索条件ごとに生成するだけにとどまっておき、我々が問題とする各要素間の依存性については考慮されていない。

Pieraccini ら^{9),10)}は、CHRONUS モデルと呼ぶ HMM を用いて、発話に対しその深層格ラベルを付与している。この HMM は、格ラベルに対応する複数の状態を有している。状態間の遷移確率は、格ラベル間の n-gram で表現され、状態からの単語出力確率は、格ラベルごとの単語 n-gram で表現されている。しかし、このモデルでは、推定対象となる意味要素が、発話中の連続する部分単語列と、sequential に、かつ、1 対 1 に対応することが前提となっており、ここで我々が対象とする Concept の推定に対して適用することはできない。

Imai ら³⁾は、放送ニュースからの話題 (集合) 抽出において、話題混合モデルを提案している。モデルは一種の HMM で、話題に対応する状態を複数有する。各状態からは、その話題に強く関係する単語を高い確率で出力する。この手法は、ニュースのような比較的長い story と各話題に顕著なキーワードが期待できる場合有効と考えられる。しかし、我々の問題は発話という比較的短い単位からの推定であり、単語と concept の間の関係は、他の単語に強く依存することが予想される。したがって、単純に単語・話題間の共起確率の形式で扱うことは難しいと考えられる。

3.2 推定モデルの定式化

単語列として与えられる発話 (SDU) から、対応す

る組合せ concept を推定する。これは、単語列から組合せ concept への識別問題として定式化できる。

発話中の単語列を W をとし、組合せ concept を $C \subset \{c_1, \dots, c_M\}$ とする。このとき、求める組合せ concept C^* は次式を満足するものとして与えられる。

$$C^* = \arg \max_{C \in \text{def}} Pr(C|W) \quad (1)$$

式 (1) に対して、これを直接モデル化することも可能であるが、実際にはデータのスパースネスの問題から信頼できるパラメータを学習することは難しい。これは、2.3 節で述べたように組合せ concept は 22,344 種類も存在し、これらすべての組合せに対して十分な量の発話データが得られることが期待できないためである。したがって、個々の concept に対し、できるだけ一般化可能な知識を抽出する必要がある。

そこで、式 (1) において、 $Pr(C|W)$ を次式のように展開し concept 単位でのモデル化を行うことで、スパースデータの問題の回避を図る。

$$\begin{aligned} Pr(C|W) &= k Pr(b_1^C|W) \cdot Pr(b_2^C|b_1^C, W) \cdots \\ &= k \prod_{i=1}^M Pr(b_i^C|b_1^C, \dots, b_{i-1}^C, W) \end{aligned} \quad (2)$$

ここで、 k は正規化定数

$$k = 1 / \sum_{C' \in \text{def}} \prod_{i=1}^M Pr(b_i^{C'}|b_1^{C'}, \dots, b_{i-1}^{C'}, W)$$

であり、 b_i^C は、Concept C に依存した c_i の解釈

$$b_i^C = \begin{cases} c_i = T & \text{if } c_i \in C \\ c_i = F & \text{otherwise} \end{cases}$$

であり、それぞれ「 c_i が表出する」「 c_i が表出しない」を表すものとする。

3.3 独立推定モデルの問題

最も簡単な推定モデルとして、concept を互いに独立として扱う方法が考えられる。すなわち、concept 間に相互非依存の仮定を置くと、式 (2) は次式のように簡略化することができる。

$$Pr(C|W) \simeq k' \prod_{i=1}^M Pr(b_i^C|W) \quad (3)$$

式 (3) は、 M 種類の concept それぞれに対し、与えられた単語列のもとでその concept が表出する確率 $Pr(c_i = T|W)$ (または表出しない確率 $Pr(c_i = F|W)$) を、他の concept とは独立に求めることに相当する。しかし、このような単純化を行った場合、以

下に述べるような問題が残る。

第1の問題点は、キーワードの階層依存性である。すでに2.3節で述べたように、Conceptの定義は、複数のconceptで構成される階層木の形式で与えられる。階層木で概念を扱うことは、比較的少数の要素(concept)により多岐にわたる概念を表現できるというメリットがある反面、階層によっては、そのconceptと表層との直接的な関係(つまり、conceptのキーワード)が曖昧になるものが出てくる。

たとえば、concept “features” は、なんらかの対象の「特徴」を表す概念要素として、DA定義階層中のいたるところに現れる。「部屋の特徴」“features + room”に関する発話では、しばしば、部屋番号を表す数詞列、部屋の大きさを表す “big”, “bigger” といった単語がこのconceptのキーワードになる。一方で、数詞列はクレジットカードに関する発話では “features” とは結び付かないし(このとき Concept は単に “payment” となる)、単語 “big” は、催し物の話題では一般的な単語としてしばしば出現する。したがって、これらの単語は concept “room” に依存した “features” のキーワードといえる。また、「団体の特徴」“features + party” に関する発話では、人数を表す数詞のほか、“me”, “only”, “with”, といった一般的な単語がキーとなることがある。これらは、“party” に依存した “features” のキーワードといえる。このように多くの階層にまたがって使用される concept の中には、自己のキーワードが階層に依存して変化するものがあり、それゆえ単独で表層との関係を学習することは難しい。

第2の問題点は、組合せ規定による concept の隠蔽である。2.3節で述べたように、組合せ concept は文生成の観点からマージが施されている。したがって、たとえある発話が「部屋の特徴」“features + room” を話題とするものであっても、その発話意図が利用可能性の言及にある場合、対応する組合せ concept は “availability + room” となり、concept “features” は隠蔽される。またこの逆に、たとえば、発話 “I’d like to reserve a room.” には、部屋や予約の “features” に言及する部分はないが、この発話の組合せ concept は “reservation + features + room” となり、“features” が誘出される。これは、文生成のテンプレートが、シングル、ツイン、といった情報を含む発話と共有されているからである。こうしたアプリアリナ制約によりその表出を直接影響される concept に対し、concept 単独で表層との関係を抽出することは難しい。

こうした問題から、concept 間の独立性を仮定するモデルでは高い信頼性は期待できず、他の concept の

存在を考慮することが望ましいと考えられる。

4. 決定木を用いた概念推定モデル

3章で述べたように、概念推定においては、他の concept の存在を考慮したモデル化が必要である。本章では、この要件を満足する概念推定モデルを提案する。式(2)における他の concept の表出条件 b_1^c, \dots, b_{i-1}^c は、他の木への参照機能により実現される。

4.1 決定木の生成

各 concept c_i に対し、 c_i が発話に付与される確率を与える決定木を生成する。以下に、そのアルゴリズムを説明する。

まず、 c_i に対応する発話事例集合を生成する。図1に、concept “availability” に対応する発話事例の一例を示す。発話事例は「属性」と「クラス」のペアである。発話の属性としては、1) 単語表層形が発話中に現れるか否か、2) ある標準形を持つ単語が発話中に現れるか否か、3) 他の concept が発話に付与されているか否か、の3種類の情報を用いる。発話のクラスは、 c_i が付与されているか ($c_i = T$)、否か ($c_i = F$)、の2クラスである。

次に、全発話事例集合を保持するルートノードを生成する。ルートノードより開始し、各ノードの事例集合 S に関し、式(4)で与える相互情報量 $I(c_i; f)$ が最大となる属性 f^* を、そのノードのテストとして採用する(ただし、親ノードですでに f^* を使っているときは、同基準で別の f を探す)。 f^* の有無で二分した S の部分集合をそれぞれ保持する新たな子ノードを生成し、子ノードの分割を繰り返すことで、木を生成する。

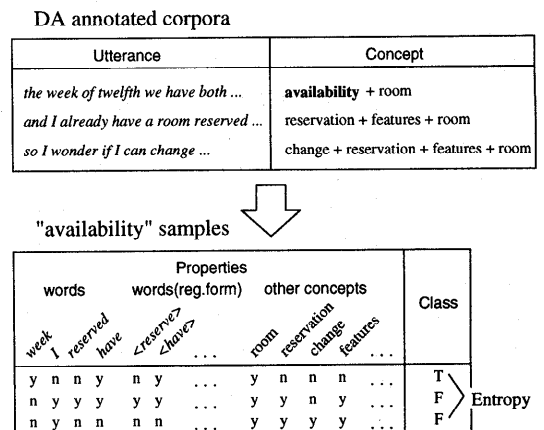


図1 “availability” 発話事例の生成
Fig. 1 Sample cases for “availability”.

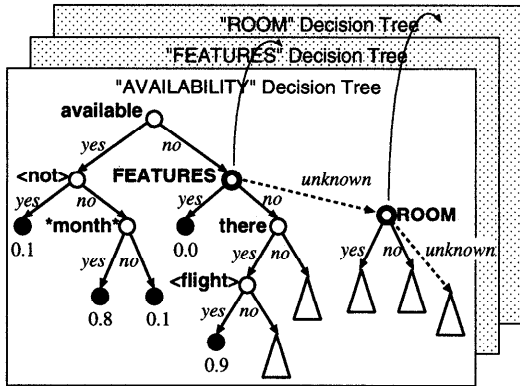


図2 相互参照を持つ決定木
Fig. 2 Decision trees with mutual reference.

$$\begin{aligned}
 I(c_i; f) = & \\
 & - \sum_{b_i} Pr(b_i|S) \cdot \log Pr(b_i|S) \\
 & + \sum_{b_i} \{Pr(b_i, f=y|S) \cdot \log Pr(b_i|f=y, S) \\
 & + Pr(b_i, f=n|S) \cdot \log Pr(b_i|f=n, S)\} \quad (4)
 \end{aligned}$$

ここで、

$$b_i \in \{c_i=T, c_i=F\} \quad (5)$$

分割は、次式で定めるノード n のエントロピーの期待値 $\tilde{H}(c_i|n)$ が閾値以下になるまで継続する。ただし、 $Pr(S)$ はノードに到達する確率であり、全学習事例に対する S の比で求める。

$$\tilde{H}(c_i|n) = -Pr(S) \cdot H(c_i|n) \quad (6)$$

ただし、

$$H(c_i|n) = - \sum_{b_i} Pr(b_i|S) \cdot \log Pr(b_i|S) \quad (7)$$

木は、基本的に f^* に対する yes/no アークによる 2 分木として生成する。ただし、 f^* として concept の有無を用いる場合は、さらに S をそのまま保持する第 3 の子ノードを生成し、unknown アークによるリンクを張る。これは、次節で後述する推定アルゴリズムにおいて、 c_i を他の concept f^* に依存せずに推定するパスを生成しておくためである。

図 2 は、このアルゴリズムによって生成される決定木の一例を示すものである。図中、白抜き丸は中間ノードであり、黒丸はリーフノードである。ノードの横に、属性テスト f を示す。属性テストのうち、英大文字からなるものは concept であり、英小文字からなるものは単語表層形、 $\langle \rangle$ で囲まれているものは単語標準形、**で囲まれているものは 5.1 節で後述する前処理により、特別なシンボルに置き換えた単語をそれぞれ表している。また、リーフノードの下の数値は、

ノードに属する事例集合のうち、concept c_i を付与されているものの割合を示している。

決定木の生成において、concept を属性として用いることにより、次のような効果が期待できる。式 (4) の相互情報量基準で信頼度の高い知識を抽出するためには、ノードに十分な数の事例が残されていることが必要である。ところで、3.3 節で述べたように、ある concept c_i の有無は、他の concept c_j の存在に依存していることがある。ここでもし、concept を属性として使わずに c_i の決定木を生成しようとするなら、この決定木では、暗に c_j の存在を調べるための属性テスト (単語) を複数生成し、事例を切り刻むことになるであろう。あるいは、 c_j の存在を十分に調べることができないであろう。concept を属性として用いることで、他の concept に関する情報集約とこれによる事例分散化の抑止がなされ、信頼度の高い知識が抽出されると期待できる。

4.2 推定アルゴリズム

本手法は、式 (2) の近似であり、concept の尤度は他の concept の表出に依存して与えられる。式 (2) における各 concept 尤度の評価順序、つまり、どの concept から先に推定するかは、発話に応じ決定木間に順序付けを行うことで決定する。決定木 T_i と T_j の間に順序付け $T_j \succ T_i$ がなされるとき、concept c_i の尤度は c_j に依存して与えられる。ただし、 c_j が有益な情報として学習されなかった場合、または、 c_j の表出に対して信頼できる尤度が得られない場合は、独立に与える。一方、 c_j の尤度はつねに c_i と独立に与えられる。

以下、図 3 を用いて、組合せ concept の推定アルゴリズムを詳細に説明する。

図 3 上段に示す 4 つの決定木 $T_1 \sim T_4$ は、それぞれ concept $c_1 \sim c_4$ の推定を行うためのものである。白抜き丸は中間ノードであり、黒丸はリーフノードである。各ノードから伸びる実線の枝は、ノードの属性テストに対する yes/no アークに対応する。破線の枝は、unknown アークであり、concept を属性テストとするノードから伸びている。ノードの横に記した「+」記号は、トレースポイントである。トレースポイントは、推定開始時にはルートノードにおかれ、属性テストの結果に応じてアークをたどることで、下位のノードへ移されていく。トレースポイントがリーフノードに到達したとき、その木に対応する concept の推定は終了する。リーフノードの下の数値が concept の推定尤度である。 T_1 により推定された $Pr(c_1=T)$ は 0.1 となっている。ただし、実験では、リーフにお

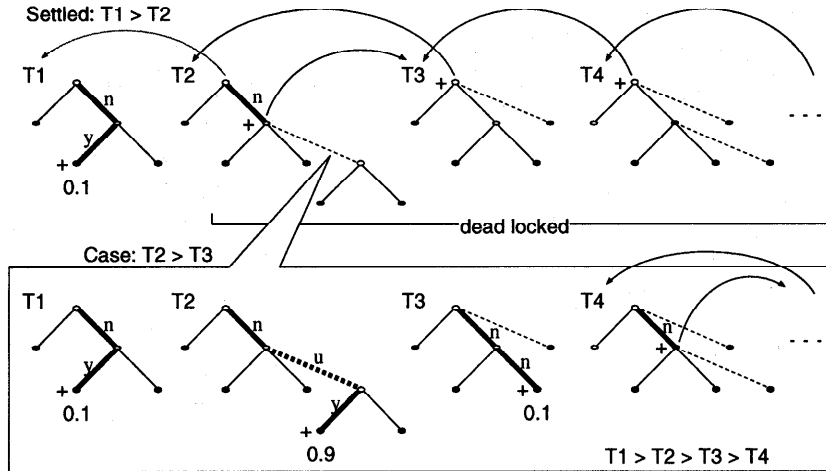


図3 動的順序付けによる相互依存推定
Fig. 3 Dynamic ordering scheme.

ける学習データの観測確率 $\overline{Pr}(c_i=T)$ を次式で平滑化して用いた ($\alpha = 0.99$ とした)。

$$Pr(c_i=T) = \alpha(\overline{Pr}(c_i=T) - 1/2) + 1/2 \quad (8)$$

図3において、弧状有向線は T_i のノードから T_j への参照であり、ノードが concept c_j の表出をテストすることを示している。このとき、以下のケースがありうる。1) T_j のトレースが終了している場合、 $Pr(c_j=T)$ の値に応じて1-a)~1-c)の決定的なトレースを行う。このとき、決定木 T_i と T_j の間に $T_j \succ T_i$ の順序付けがなされる。図3上段の例では $T_1 \succ T_2$ である。1-a) もし、 $Pr(c_j=T) > 1 - \theta$ なら c_j の表出を仮定し、yesのアークをたどる。これは、 c_i の尤度を $c_j = T$ の条件付き確率で求めることに相当する。1-b) $Pr(c_j=T) < \theta$ なら c_j の非表出を仮定し、noのアークをたどる。これは、 c_i の尤度を $c_j = T$ の条件付き確率で求めることに相当する。1-c) $\theta \leq Pr(c_j=T) \leq 1 - \theta$ なら、 c_j は曖昧であるとし、unknownのパスをたどることにより、 c_j とは独立に推定する。一方、2) もし、 T_j のトレースがまだ終了していないなら、トレースを一時中止し、 T_j のトレースが終わるのを待つ。

決定木群のトレースは、並列に行う。ただし、決定木がお互いに依存し合いデッドロックの状態に陥った場合(図3上段の例では、 T_2 以右の木がデッドロックしている)、後述する基準によりただ1つ選択した T_{i^*} のトレースポイントを unknown アーク下のノードへと移動させ、デッドロックしていたすべての木のトレースを再開する。このとき、 T_{i^*} が参照していた木を T_{j^*} とすると、 $T_{i^*} \succ T_{j^*}$ の順序付けがなされる。デッドロック時に進める木 T_{i^*} は、 T_{i^*} の各仮説の

もとで暫定的なトレースを行い、評価することで決定する。デッドロック中の木を $T_i \in D$ とする。各 T_i について、暫定的に T_i のトレースポイントを unknown アークの下へと移動させ、デッドロックしていたすべての木のトレースを再開してみる。図3下段は、 $T_i = T_2$ の例である。 T_2 のトレースポイントを動かして T_2 以右の木のトレースを再開すると、 T_2 はリーフまで到達し、この結果 T_2 に依存していた T_3 もリーフに到達する。 T_4 は下位のノードに進むが、そこで他の木と新たなデッドロックに陥り停止する。また、この過程で $T_2 \sim T_4$ の間にも $T_2 \succ T_3 \succ T_4$ なる順序付けがなされる。

仮説 $i^* = i$ における暫定的なトレースで到達する $T_j \in D$ のノード(リーフでなくともよい)を $n_j^{(i)}$ で表すとき、次式で与えられる T_{i^*} をデッドロックの解消に最適なものとして選択する。これは、全 concept の推定における曖昧さを、次のデッドロックが生じるまでに最小にするものである。

$$T_{i^*} = \arg \min_i \sum_{j \in D} H(c_j | n_j^{(i)}) \quad (9)$$

すべての木のトレースが終了したとき、求める組合せ concept C^* は次式で与えられる。

$$C^* = \arg \max_{C \in \text{def}} \prod_{i=1}^M Pr(c_i | T_i) \quad (10)$$

本手法が式(2)の近似を意図するものであることは、本節の冒頭で述べた。ところで、式(2)を忠実に求めるには、 c_i の確率を c_j ($1 \leq j \leq i-1$) のあらゆる表出/非表出条件のもとで与えることが必要である。一方、本手法では、 c_i の確率をつねに c_j ($1 \leq j \leq i-1$)

の最尤の表出/非表出条件のもとからしか与えていない(あるいは、一部の c_j を無視する)。したがって、この近似は必ずしもすべての C に対して正しい確率を与えていない。

しかし、各決定木に式(11)の楽観的な仮定、すなわち、閾値を超えて致命的な過ちを犯すことはないことを仮定するとき、本手法は式(2)の近似として正しい推定を行うことが可能である。

$$\forall c_i Pr(b_i^{\hat{C}} | b_1^{\hat{C}}, \dots, b_{i-1}^{\hat{C}}, W) \geq 0 + \theta \quad (11)$$

ただし、 \hat{C} は正解の組合せ concept を表す。

5. 評価実験

5.1 実験条件

表4に示すCSTARのDAタグ付き英語テキストコーパスを用い、提案手法による組合せ concept の推定実験を行った。データは模擬会話の書き起こしテキストであり、SDUへの分割とDAタグの付与は人手で行われている。CMUとETRIにより作成された56会話を用いた。タスクは旅行会話であり、これにはホテル予約、フライト予約、観光案内、旅行パックの問合せ、等が含まれている。実験では前処理により、ホテル名・人名等の固有名詞や、曜日名、数字列、等の単語(列)を特別なシンボルで置き換えている。

本手法による組合せ推定の効果を確認するため、実験は提案手法(以下、DTdependと表記する)のほか、以下の2種類の決定木を用いて実験を行った。

- conceptごとの独立推定(DTindep)

各 concept に対応する複数の決定木を用いる。決定木間の相互参照機能を持たない点が提案手法との違いである。これは、式(3)のモデル化に相当する。

- 組合せ concept の一括推定(DTlump)

組合せ concept C を直接識別するよう学習した、ただ1つの決定木を用いる。これは、式(1)のモデル化に相当する。推定は次式で行う。

$$C^* = \arg \max_C Pr(C|T)$$

5.2 実験結果

表5に、3手法による組合せ concept の推定精度を示す。

提案手法(DTdepend)では、クローズドテスト、オープンテストともに最良の正解率が得られていることが分かる(クローズドでは、DTindepと同等)。

オープンテストにおいて、提案手法では、DTindepと比較して正解率2.0%(誤り改善率11%)の改善が

表4 実験データ
Table 4 Experimental data.

	クローズド	オープン	合計
発話(SDU)	1,497	499	1,996
語彙(前処理後)	729 (534)	509 (365)	807 (589)
concept	42	36	43
組合せ concept	82	59	93

表5 組合せ concept の推定精度
Table 5 Accuracy for Concept (combination).

正解率(%)	DTdepend	DTindep	DTlump
クローズド	99.6	99.6	97.4
オープン	84.0	82.0	77.6

DTdepend : 提案手法

DTindep : 複数決定木による独立推定

DTlump : 単一決定木による一括推定

見られる。両手法の相似性から、これは提案手法で concept の表出/非表出を利用した効果であると考えられる。

DTlumpでは、他の2手法より大きく劣る結果となった。DTindepと比較した誤り改善率は、クローズドで-550%、オープンで-24%である。これは、3.2節で予想した組合せのスパースネスの問題を顕著に表していると考えられる。

5.3 考察

本稿で提案する手法は、1) conceptごとの決定木生成、2) 決定木間の相互参照機能、の2点を特徴とするものであった。本節では、実験結果よりそれぞれの効果を考察する。

conceptごとに決定木を生成する狙いは、3.2節で述べたように、組合せのスパースネスを回避することであった。提案手法DTdependおよび比較手法DTindepは、conceptごとに知識を抽出する方式であるため、特に少量のサンプルしか得られない組合せ(Concept)に対して、比較手法DTlumpより安定した推定が可能であると期待される。

そこで、オープンデータ499発話から、特に学習データの少ないConceptが付与された発話を選び、その正解率を調べた。表6に結果を示す。母集団が小さいため厳密な比較は難しいが、表6より、学習データにおける出現頻度が1回未満/10回未満ときわめて少ない組合せでは、提案手法DTdependおよびDTindepと、DTlumpとの精度差が大きくなっていることが分かる。特に、学習データ中にまったく現れない組合せ(出現頻度が1回未満)の11発話に対し、提案手法DTdependおよびDTindepでは、それぞれ4発話、3発話を正しく推定することができた。一方、

表6 学習データが少ない Concept の正解率

Table 6 Accuracy for Concepts which have few training data.

学習データ 中の頻度	評価データ 中の発話	オープンテスト正解率 (%)		
		DTdepend	DTindep	DTlump
1 回未満	11	36.4	27.3	0.0
10 回未満	61	50.8	47.5	31.1
100 回未満	262	72.5	69.5	63.4

DTlump は学習データに出現する組合せへ直接分類する手法であるため、こうした未知の組合せを推定することはできない。このようなスパースデータの問題は、学習データを増やすことでも改善が期待できるものの、人手による意味タグの付与はコストが高く、また、特定ドメインにおいて収集可能な言語データ量には限界があることから、本手法の有効性が期待される。

次に、相互参照機能の有効性について考察する。決定木間に相互参照機能を持たせた狙いは、スパースデータ回避の観点から concept 単位とした決定木において、個々の決定木に他の concept の表出を最大限利用させることにあった。一方、表5の結果からは、提案手法 DTdepend の正解率 84.0%は、DTindep の正解率 82.0%を上回ったものの、期待するような大きな精度向上は見られなかった。

しかし、両手法の正解結果を観察してみると、比較手法 DTindep において、複数の Concept が識別不能のまま同率 1 位候補としてあげられている例が多く見られた。表7上段に、オープンテストで複数候補により正解となった発話数を示す。こうした複数候補は、個々の concept の尤度が正しく与えられなかった場合に生じる。

たとえば、concept の脱落誤りをともなう複数候補正解の例として、発話 “the singles I mean they have a king size bed” があげられる。この正解 Concept は “features + room” である。この発話に対し、DTindep は各 concept の尤度を以下のように推定しており、正解 concept である “features” に脱落誤りを生じていた。ところが組合せ規定より “room” を単独で用いることは許されていないため、定義中のすべての “X + room” の組合せ (“availability + room”, “location + room”, “price + room”, etc.) が同率 1 位候補としてあげられていた。

- “room” = 1.0
- その他の concept = 0.0

また別の例として、concept の挿入誤りをともなう複数候補正解の例をあげる。発話 “doubles are fifty dollars a night” の正解 Concept は “price + room”

表7 複数候補による正解発話数 (オープン)

Table 7 Number of utterances correctly identified in multiple candidates.

	DTdepend	DTindep	DTlump
複数候補の正解発話	15	36	5
単独候補の正解率 (%)	81.0	74.7	76.6

である。この発話に対し、DTindep は各 concept の尤度を以下のように推定しており、concept “features” の挿入誤りを生じていた。そのため、組合せとしては “price + room” と “features + room” が同率 1 位候補としてあげられていた。

- “price”, “room”, “features” = 1.0
- その他の concept = 0.0

両例において DTindep が推定を誤った “features(+room)” 「(部屋の) 特徴」は、“price + room” 「部屋の値段」や “availability + room” 「部屋の利用可能性」といった発話と表層が似通っているため、“features” 単独での学習/推定が困難だったと思われる。一方、提案手法 DTdepend では、両例いずれに対してもすべての concept を正しく推定しており、結果、正解の組合せのみを 1 位候補としてあげていた。これは、提案手法が他の concept との競合をうまく解消できたためと考えられる。

表8に、concept 単独での推定誤差を示した。DTindep で誤りがあった concept 31 種類のうち、約 1/3 にあたる 11 種類において提案手法による改善が見られ、また、提案手法で悪くなった concept はなかった。表7下段には、複数候補による正解を除外した正解率を示した。提案手法 DTdepend では他手法と比較して、大きく精度が向上していることが分かる。

5.4 混合モデルによる比較実験

従来手法として、3.1 節で引用した話題混合モデル³⁾を用い、比較実験を行った。各混合モデルはいずれかの組合せ concept に対応している。それぞれのモデルは、concept に対応する複数の状態と、いずれの concept にも対応しない一般的な単語を出力する状態を有する。これら状態の出力確率は、モデル間で共有される。実験データは、表4と同じ前処理を行った単語系列を用いた。混合モデルの初期値には、出力確率、遷移確率、ともに一様分布を与え、EM アルゴリズムで学習データに対する尤度が収束するまで、10 回の繰返し学習を行った。

このときの混合モデルによる Concept 正解率は、クローズド 91.5%、オープン 76.5%であった。この値は提案手法の正解率 (表5:クローズド 99.6%、オープン 84.0%) を下回る。特に、クローズドテストにおいて

表 8 concept 単独でみた推定誤差の改善
Table 8 Improvement of errors for each concept.

concept	平均推定誤差		改善率 (%)	concept	平均推定誤差		改善率 (%)
	DTdepend	DTindep			DTdepend	DTindep	
features	0.058	0.091	36.3	preference	0.020	0.020	0.0
null	0.048	0.057	15.3	numeral	0.016	0.018	11.1
temporal	0.045	0.053	15.2	room	0.018	0.018	0.0
reservation	0.024	0.034	28.4	party	0.014	0.015	6.7
price	0.024	0.032	25.0
availability	0.030	0.030	0.0	合計	0.009	0.011	15.1

concept は DTindep の推定誤差が大きい順に 20 個を表示した。

も両者の精度差が大きくなっていることから、単語間の非依存性を仮定する混合モデルでは、発話 (SDU) のような比較的短い単位からの Concept 抽出を学習できなかったのではないかと考えられる。

6. む す び

自然発話から中間言語への頑健な変換方式を実現するため、特に発話の概念を対象とした検討を行った。発話の概念は複数の項の組合せで表現される。この組合せにはアプリアリな規定が存在し、それゆえ推定時の問題として、項の間に相互非依存の仮定を置くことが難しいことを述べた。これら項の相互依存性の問題に対し、相互参照機能を持つ決定木群の動的順序付けにより、発話の概念を高精度に推定する方式を提案した。また、英語対話コーパスを用いた実験では、各項を独立に推定する手法と比較して 11% の誤り改善率が得られた。今後、中間言語の他の要素である、Speech act や Argument の抽出を検討する必要がある。また、本方式を拡張し、音声認識への適用を検討する必要がある。

謝辞 本研究の機会を与えてくださった、ATR 音声翻訳通信研究所山本誠一社長に感謝する。また、本研究を進めるにあたって適切な助言をいただいた、ATR 音声翻訳通信研究所第 1 研究室の方々に感謝する。

参 考 文 献

- 1) 谷垣宏一, 政瀧浩和, 匂坂芳典: 決定木を用いた発話の意味タグ推定, 日本音響学会平成 10 年度春季研究発表会論文集, No.1-6-2, pp.3-4 (1998).
- 2) Kuhn, R. and Mori, R.: The application of semantic classification trees to natural language understanding, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.17, No.5, pp.449-460 (1995).
- 3) Imai, T., Schwartz, R., Kubala, F. and Nguyen, L.: Improved topic discrimination of broadcast news using a model of multiple simultaneous topics, *Proc. ICASSP-97*, pp.727-730 (1997).
- 4) <http://www.is.cs.cmu.edu/cstar/>
- 5) Angelini, B., Cettolo, M., Corazza, A., Falavigna, D. and Lazzari, G.: Multilingual person to person communication at IRST, *Proc. ICASSP-97*, pp.91-94 (1997).
- 6) Woszczyna, M. and Waibel, A.: Inferring linguistic structure in spoken language, *Proc. ICSLP-97*, pp.847-850 (1994).
- 7) Schwartz, R., Miller, S., Stallard, D. and Makhoul, J.: Hidden understanding models for statistical sentence understanding, *Proc. ICASSP-97*, pp.1479-1482 (1997).
- 8) Papineni, K., Roukos, S. and Ward, R.: Maximum likelihood and discriminative training of direct translation models, *Proc. ICASSP-98*, pp.189-192 (1998).
- 9) Levin, E. and Pieraccini, R.: CHRONUS, The next generation, *Proc. ARPA Spoken Language Technology Workshop*, pp.269-271 (Jan. 1995).
- 10) Pieraccini, R., Tzoukermann, E., Gorelov, Z., Gauvain, J.L., Levin, E., Lee, C.H. and Wilpon, J.G.: A speech understanding system based on statistical representation of semantics, *ICASSP-92*, Vol.1, pp.193-196 (1992).
- 11) Lavie, A., Levin, L., Zhan, P., Taboada, M., Gates, D., Lapata, M., Clark, C., Broadhead M. and Waibel, A.: Expanding the domain of a multi-lingual speech-to-speech translation system, *Proc. Workshop on Spoken Language Translation, ACL/EACL-97* (July 1997).
- 12) Epstein, M., Papineni, K., Roukos, S., Ward, T. and Della Pietra, S.: Statistical natural language understanding using hidden clumpings, *Proc. ICASSP-96*, pp.176-179 (1996).
- 13) Minker, W., Bennacef, S. and Gauvain, J.: A stochastic case frame approach for natural language understanding, *Proc. ICSLP-96*, pp.1-4 (1996).
- 14) 市川, 荒木, 石崎, 板橋, 伊藤, 柏岡, 加藤, 熊谷, 樽松, 小磯, 田本, 土屋, 中里, 堀内, 前川,

山下, 吉村: 談話タグ標準化の現状, 人工知能学会言語・音声理解と対話処理研究会(第21回)資料, SIG-SLUD-9703-7, pp.41-48 (1998).

- 15) Jokinen, K. and Morimoto, T.: Topic information and spoken dialogue systems, *Proc. NLPRS-97*, pp.429-434 (1997).
- 16) Sakamoto, H. and Matsunaga, S.: Continuous speech recognition using a dialog-conditioned stochastic language model, *Proc. ICSLP-94*, pp.811-814 (1994).

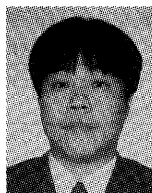
(平成10年10月15日受付)

(平成11年2月8日採録)



谷垣 宏一 (正会員)

平成4年東北大学工学部情報工学科卒業。平成7年同大学大学院情報科学研究科修士課程修了。同年三菱電機(株)情報技術総合研究所音声言語インタフェース技術部入社。平成9年よりATR音声翻訳通信研究所に出向。自然言語処理, 音声認識の研究に従事。日本音響学会会員。



政瀧 浩和

平成元年京都大学工学部電子工学科卒業。平成3年同大学大学院工学研究科修士課程修了。同年住友金属工業株式会社システムエンジニアリング事業本部入社。平成7年より平成10年までATR音声翻訳通信研究所に出向。平成10年より住友金属工業(株)システム事業部勤務。自然言語処理, 音声認識の研究に従事。電子情報通信学会, 日本音響学会各会員。



匂坂 芳典

昭和48年早稲田大学理工学部物理学科卒業。昭和50年同大学大学院修士課程修了。同年日本電信電話公社(現, NTT)武蔵野電気通信研究所入社。昭和61年より国際電気通信基礎技術研究所(ATR)に出向。現在, ATR音声翻訳通信研究所第1研究室室長。工学博士。音声合成・音声認識を中心とした, 音声情報処理, 言語情報処理の研究に従事。電子情報通信学会, 日本音響学会, IEEE, 米国音響学会各会員。